Volume 3 Nomor 3 September 2018

INFORMASI INTERAKTIF

JURNAL INFORMATIKA DAN TEKNOLOGI INFORMASI

PROGRAM STUDI TEKNIK INFORMATIKA – FAKULTAS TEKNIK -UNIVERSITAS JANABADRA

INFORMASI INTERAKTIF	Vol. 3	No. 3	Hal. 154 - 209	Yogyakarta September 2018	ISSN 2527-5240

INFORMASI INTERAKTIF

ISSN 2527-5240

DEWAN EDITORIAL

Alamat Redaksi : Program Studi Teknik Informatika Fakultas Teknik
Universitas Janabadra
Jl. Tentara Rakyat Mataram No. 55-57
Yogyakarta 55231
Telp./Fax : (0274) 543676
E-mail: informasi.interaktif@janabadra.ac.id
Website : http://e-journal.janabadra.ac.id/

Frekuensi Terbit : 3 kali setahun

JURNAL INFORMASI INTERAKTIF merupakan media komunikasi hasil penelitian, studi kasus, dan ulasan ilmiah bagi ilmuwan dan praktisi dibidang Teknik Iinformatika. Diterbitkan oleh Program Studi Teknik Informatika Fakultas Teknik Universitas Janabadra di Yogyakarta, tiga kali setahun pada bulan Januari, Mei dan September.

ISSN 2527-5240

DAFTAR ISI

ISSN 2527-5240

PENGANTAR REDAKSI

Puji syukur kami panjatkan kehadirat Allah Tuhan Yang Maha Kuasa atas terbitnya JURNAL INFORMASI INTERAKTIF Volume 3, Nomor 3, Edisi September 2018. Pada edisi kali ini memuat 8 (delapan) tulisan hasil penelitian dalam bidang teknik informatika.

Harapan kami semoga naskah yang tersaji dalam JURNAL INFORMASI INTERAKTIF edisi September tahun 2018 dapat menambah pengetahuan dan wawasan di bidangnya masing-masing dan bagi penulis, jurnal ini diharapkan menjadi salah satu wadah untuk berbagi hasil-hasil penelitian yang telah dilakukan kepada seluruh akademisi maupun masyarakat pada umumnya.

Redaksi

APLIKASI PENGAMANAN WEB

Indra Yatini B[1], F. Wiwiek Nurwiyati[2], Ikhwan Dirga Pratama[3]

[123]Teknik Informatika, STMIK AKAKOM Yogyakarta

Email :[1]indrayatini@akakom.ac.id, [2]wiwiek@akakom.ac.id , [3]agrid@outlook.com

ABSTRACT

Security weaknesses in the web application are often utilized by people who are not responsible for stealing data and damaging the web, which of course this will harm web owners and interfere with the convenience of web users. It takes a way to reduce the risk of attacks on web applications that have security weaknesses. One of them is by using the Web Application Firewall. The Web Application Firewall in this study was designed to secure web applications from brute force attacks, SQL Injection, XSS, Comman Execution, and Arbitrary File Upload. This research was conducted by comparing the results obtained from testing the dummy website that has been installed with the Web Application Firewall using the PHP programming language.

Keywords: *Arbitrary File Upload, Brute Force, Command Execution, PHP, XSS, Web Application Firewall*

1. PENDAHULUAN

Perkembangan teknologi world wide web (www) pada era sekarang ini sudah sangat berkembang dengan pesat, sehingga memunculkan media baru di berbagai aspek dalam penyebaran informasi dan peningkatan komunikasi dimasyarakat seluruh dunia. Barbagai aktivitas yang sebelumnya dinilai tidak mungkin dapat dilaksanakan sekarang telah menjadi bagian dari masyarakat teknologi terkini. Tukar menukar surat (*surel*) dan juga keberadaan dari halaman web adalah bentuk komunikasi yang membawa perubahan besar pada kehidupan manusia. Dengan adanya teknologi web, sepertinya dunia menjadi tanpa batas dengan potensi pengembangan yang tidak ada batasnya [1].

Fungsi – fungsi web secara umum adalah sebagai komunikasi, informasi, hiburan dan transaksi. Dengan fungsi – fungsi web tersebut maka keamanan web harus dijaga dengan benar untuk memperlancar komunikasi, menjaga keamanan dan kerahasiaan informasi serta kenyamanan dan keamanan dalam bertransaksi melalui media website. Sehingga dengan begitu akan tercapai fungsi- fungsi dari sebuah website.

Seiring perkembangan web yang demikian pesat maka keamanan web harus di jaga dengan benar untuk memperlancar kumunikasi, menjaga keamanan dan kerahasiaan informasi serta kenyamanan dan keamanan dalam bertransaksi melalui media website. Sehingga dengan begitu akan tercapai fungsi-fungsi dari sebuah website [2].

Kejahatan di dunia teknologi dan informasi terutama pada aplikasi web semakin marak terjadi. Salah satu factor yang menyebabkan kurangnya tingkat keamanan pada aplikasi web adalah kesalahan penulisan kode program. Kesalahan penulisan kode program dalam pembuatan aplikasi web adalah hal yang sering dimanfaatkan oleh para penyerang, hal ini mengakibatkan rata-rata aplikasi web bisa diserang dengan memanfaatkan kesalahan ini [3]. Kelemahan – kelemahan yang sering dimanfaatkan oleh para penyerang diantaranya adalah kelemahan terhadap SQL Injection, XSS, Remote File Inckusion, dan Username Enumeration.

Salah satu teknik untuk melindungi website dari serangan peretas salah satunya adalah dengan memasang *Web Application Firewall*. Ada banyak bahasa pemrograman yang bisa digunakan untuk membuat *Web Application Firewall***s**alah satu bahasa pemrograman yang populer untuk itu adalah PHP.

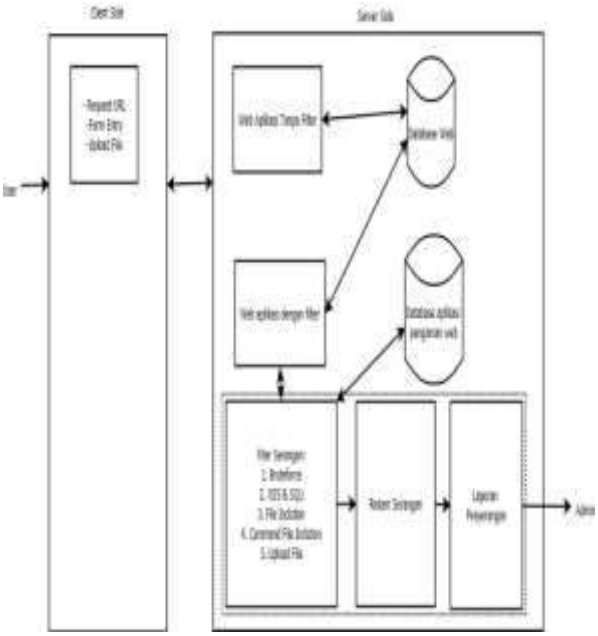
Web Aplication Firewall ini sangatlah penting digunakan untuk pengguna website atau pengembang website untuk memberi keamanan tambahan pada website yang telah dikembangkan [4].

Web Application Firewall adalah suatu metode untuk pengamanan pada aplikasi web, yang berupaya mencegahan dan ancaman dari *attacker* [5]. Web Application Firewall umumnya berjalan pada layer aplikasi yang memonitori dan memodifikasi http request, padadasarnnya *Web Application Firewall* .
Berikutdasarterori yang digunakan :

1. PHP
   PHP Merupakan Bahasa pemrograman *server-side scripting* yaitu sintaks dan perintah yang dijalankan di server dan disertakan pada dokumen HTML. Sehingga dapat digunakan untuk membuat halaman web dinamis.
2. JSON
   JSON (JavaScript Object Notation) adalah format pertukaran data yang rinngan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat oleh komputer.
3. Htmlentities
   Htmlentities adalah suatu fungsi PHP yang berfungsi untuk mengubah karaker menjadi HTML entities.
4. Escapeshellarg
   Escapeshellarg adalah fungsi PHP yang digunakan untuk menambah tanda petik pada *string* yang memungkinkan mengakses langsung perintah *shell*
5. Bruteforce
   Bruteforce adalah suatu metode untuk bisa masuk kesuatu situs web dengan menggunakan nama pengguna dan kata sandi secara acak.
6. SQL Injection
   SQL Injection adalah jenis serangan yang memungkinkan penyerang untuk memanipulasi perintah SQL melalui URL atau isian *form* yang dikirimkan oleh aplikasi web ke *server*
7. Cross Site Scripting
   Kelemahan *Cross Site Scripting* atau XSS terjadi ketika aplikasi mengambil data yang tidak dapat dipercaya dan mengirimnya ke suatu web *browser* tanpa validasi yang memadai.
8. Command Execution
   Command execution merupakan celah keamanan yang memungkinkan penyerangan untuk menyisipkan perintah-perintah shell untuk dieksekusi oleh web server.
9. Arbitrary File Upload
   Arbitrary file upload adalah celah keamanan dimana fitur upload tidak membatasi extensi apa saja yang diizinkan.

Data yang akan dipakai menggunakan *web dummie* untuk percobaan dan data string yang berbahaya guna dilakukan filter seperti (‘ / = or ;) yang didapatkan pada situs yang dikelola oleh OWASP.

Metode yang digunakan dalam perancangan system ini dengan cara menambahkannya kedalam web aplikasi yang ingin diamankan. Kemudian sistemakan menetralkan *request – request* yang membahayakan web aplikasi. Untuk lebihjelasnya akan diuraian dari arsitekur system pengaman web ini, adapun rancangan system pengamanan Web yang akandibuatdapat di lihat pada gambar 1 berikut ini.



Gambar 1. Gambaran Umum Sistem yang akan dibangun

Pada gambar 1 menunjukan gambaran umum sistem yang akan dibangun yaitu dengan cara menambahkan*Web Application Firewall* kedalam aplikasi web yang ingin diuji. Kemudian system akan menetralkan permintaan-permintaan yang membahayakan sistem.

2.1 Client Side

Pada bagian ini user sebagai pengguna web akan melakukan interaksi dengantiga cara berikut:

Request URL : *Requesturl* dilakukan oleh *user* dengan cara mengetikan url alamat website pada *browser* ataupun dengan mengklik link yang berisi url alamat sebuah website.

Form Entry : Terjadi ketika *user* mengirimkan data yang telah diisikan kedalam form sebuahsitus.

File Upload : User mengirim data berupa*file*ke server website melalui fitur *upload file* yang disediakan

a. Server Side

Website tanpa aplikasi pengaman web : Pada kondisi ini jika website memiliki celah maka data berbahaya yang dikirim user akan tereksekusi oleh aplikasi dan juga database.

Website menggunakan aplikasi pengaman web : Komunikasi data antara user, aplikasi dan *database* akan dicekter lebih dahulu oleh aplikasi pengaman sehingga jika terdapat data berbahaya akan dinetralkan terlebih dahulu oleh aplikasi pengaman.

Sistem database aplikasi pengaman web : Digunakan untuk menyimpan data *cheat sheet* celah keamanndan juga data setting aplikasi pengaman web.

Program penulisan log : Serangan – serangan yang terjadi ditulis kedalam file log.

Laporan data serangan : Dari log yang telah di buat laporan dalam bentuk chat dan map, untuk memudahkan admin dalam memonitor serangan

2. TINJUAUN PUSTAKA

Terdapat referensi penelitian sebelumnya telah dilakukan oleh [6] dengan hasil website dengan code yang aman berdasarkan OWASP TOP 10 dan[7] yang menghasilkan website yang bias menangkal serangan*SQL Injection*.

Sistem yang dibuat ini merupakan sistem berbasis web dimana pengguna harus menyisipkan salah satu berkas php kedalam situs web yang akan di pantau, dan melakukan

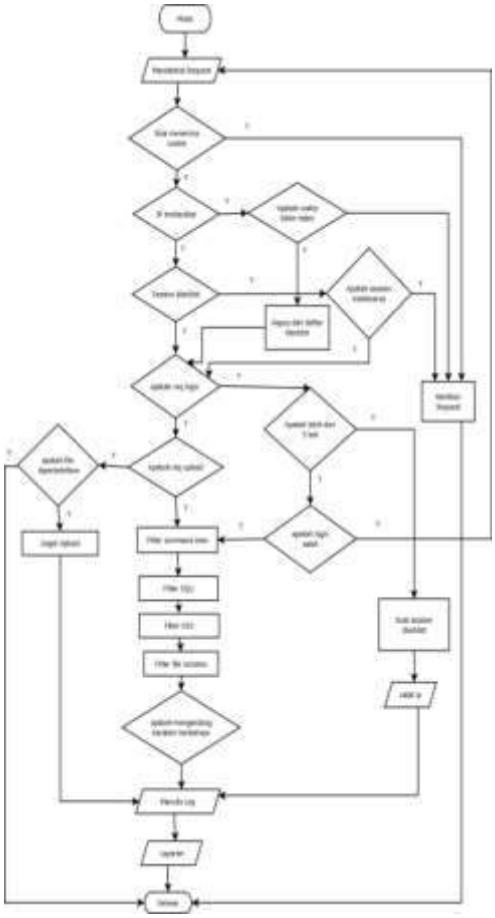
login untuk memantau jalannya aplikasi pengaman web ini.

Pembeda dari aplikasi lainnya adalah:

1) Adanya Laporan yang menggunakan Informasi Geografis untuk menampilkan IP penyerang berdasarkan lokasi negara.
2) Perbaikan UI dan UX menjadi lebih responsif.
3) Memberikan keamanan pada kode program dengan pemberian enkripsi base64.

3. METODE PENELITIAN

Dalam perancangan sustem ini menjelaskan tentang pemodelan system serta perancangan antarmuka sistem. Pemodelan sistem menggunakan flowchart dapat di lihat pada gambar 2 berikut ini.



Gambar 2 Flowchart

Gambar 2 menunjukan keseluruhan sistem yang akan dibuat.

4. IMPLEMENTASI & PEMBAHASAN

Dilakukan pengujian terhadap aplikasi yang di buat. Sebagai kasus uji (*dummies*) digunakan aplikasi web yang memiliki celah keamanan. Pengujian dilakukan pada dua web aplikasi yang sama, dimana salah satunya telah

menggunakan aplikasi pengaman dan yang lainnya tidak. Dari hasil pengujian terhadap kedua web tersebut akan dibuat perbandingan.

Pengamanan dari serangan *brute forcing* dilakukan dengan cara member batasan kepada user dalam melakukan permintaan *login*. Setiap permintaan akan ditandai dengan *session* yang akan di *increment*. Jika jumlah permintaan melebihi yang telah ditentukan dalam jangka waktu tertentu maka permintaan selanjutnya akan ditolak. Hal ini bertujuan agar tidak ada percobaan untuk melakukan *login* menggunakan semua kunci yang memungkinkan.

Proses masuk di tandai menggunakan *session count.* Jika pengguna belum memiliki *session count* maka akan di buat *session count* yang bernilai 1 (`$_SESSION['count']=1)` dan jika telah memiliki *session count* maka nilai *session* tersebut akan di *increment* (`$_SESSION['count'] +=1)`. Pada baris ke-11 dilakukan pengecekan nilai *session count*, jika nilainya melebihi batas yang telah ditentukan makan permintaan akan diblokir dan jika belum melebihi maka nilai *session count* akan di *increment*. Pada kode program di atas terdapat pemaggilan fungsi `$this -> TulisLog("BruteForce" , $this -> ambilIP())` yang berfungsi untuk memanggil fungsi menulis log.

Untuk melakukan pengamanan dari serangan *SQL Injection* dan *Cross Site Scripting* dilakukan pengubahan karakter-karakter yang biasa digunakan dalam tag-tag pemrograman dalam *query* sql kedalam karakterentitas dengan tujuan agar tag-tag pemrogramanter sebut tidak tereksekusi oleh browser, dan*query* sql yang dimasukan hanya terbaca sebagai*string* normal oleh mysql *server* dan bukanmerupakan *query*.

Fungsi *htmlspesialchars* digunakan untuk merubah tag html menjadi karakter entitas atau karakter biasa agar tidak dieksekusi oleh broser.

Pengamanan celah *command execution* ini dilakukan dengan cara memfilter perintah shell menggunakan *escape shell cmd* dengan tujuan agar jika ditemukan karakter-karakter yang bias digunakan untuk melakukan perintah ganda akan difilter sehingga perintah *shell* yang disisipkan tidak bias tereksekusi oleh*web server*.

Untuk melakukan pengamananfitur upload diperlukan nama-nama *extension file* yang diizinkan untuk *upload* kedalam aplikasi web. Dari data tersebut aplikasi pengaman web ini akan membandingan dengan *extention file* yang diunggah. Jika *extention file* yang diunggah tidak ditemukan dalam *extention file* yang diizinkan maka aplikasi akan mengagalkan proses unggah dengan tujuan agar data yang tidak diinginkan tidak masuk kedalam web aplikasi. Kemudian membuat log ataskejadian tersebut.

Dari permintaan-permintaan yang mencurigakan akan ditulis kembali kedalam log *text file* sehingga bias digunakan untuk melihat histori serangan yang telah terjadi. Data yang ditulis dalam log adalah jenis serangan, alamat ip penyerang, tangga serangan, jam serangan, parameter yang diserang.

Dari log yang telah tersimpan akan dibuat laporan dalam bentuk map yang bertujuan untuk mengetahui lokasi ip penyerang dan memberikan pengalaman UI dan UX yang lebihbaik. Map yang digunakanmenggunakan API google Maps.

Enkripsi di butuhkan untuk menambah keamanan pada file konfigurasi yang dibutuhkan. Enkripsi yang digunakan adalah enkripsi mcrypt_rijndael_256 dengan kunci 32bit.

5. HASIL PENELITIAN

Pada uji aplikasi pengaman ini dapat mengenali serangan SQL Injection, XSS, Arbitrary File Upoad, dan Command Execution dengan cara mencocokan string request.

1. Pengujian Pengaman Celah *Brute Force*
   Pengujian dilakukan pada web dummies yang telah sengaja di buat memiliki celah. Dallam pengujian ini maksimal request yang diperbolehkan adalah sebanyak 5 kali. Dari hasil pengujian sukses dan serangan Brute Force berhasil di blok.
2. Pengujian Pengaman Celah *SQL Injection*
   Untuk memperoleh hasil yang akurat maka pengujian ini menggunakan metode manual dengan cara memberikan parameter SQLi pada form yang tersedia dan mendapatkan hasil pengujian, dapat di lihat pada tabel 1 berikut ini.

Tabel 1 Hasil Dari Pengujian *SQL Injection*

Jenis	Hasil	Waktu Rata-Rata
Tanpa aplikasi pengaman web	Muncul tampilan galat SQLi	0,0026
Menggunakan aplikasi pengaman web	inputan di tolak	0,0030

Dari hasil pengujian tersebut dapat disimpulkan bahwa modul pengaman ini dapat diandalkan untukmengamankan web dari serangan SQL Injection karena pada pengujian ini 100% web dummies dapat diselamatkan dari serangan SQL 29 Injection, sedangkan sebelum terpasang modul pengaman tingkat keselamatan dari serangan SQL Injection hanya 0%, aplikasi pengaman akan menulis log serangan yang terjadi dan berikut contoh sebagian dari log serangan yang terjadi

"XSS/SQLi|36.81.17.51|2017-07-16|16:43:23|/Web-Firewall/index.php?vo=\"> "

3. Pengujian Pengaman Celah XSS
   Pengujian dilakukan pada web *dummies* dan di dapatkan hasil, dapay di lihat pada tabel 2 berikut ini.

Tabel 2.HasilPengujian XSS

Jenis	Hasil	Waktu Rata Rata
Tanpaaplikasi pengaman web	XSS berhasilterek sekusi	0,0045
Menggunakan aplikasipenga manweb	inputan di tolak	0,0075

Hasil yang ditunjukan menggambarkan bahwa aplikasi pengaman mampu menangkal serangan XSS dengan tingkat keberhasilan mencapai 100%. Dan berikut contoh sebagian log yang di dapat

"XSS/SQLi|36.81.17.51|2017-07-16|16:43:23|/Web-Firewall/index.php?vo=\"><script>alert(document.cookie);</script>"

4. Pengujian Pengaman Celah *Command Execution*
   Pengujian ini dilakukan pada web *dummies* yang terdapat fasilitas ping, dari pengujian di dapatkan hasil dapat di lihat pada tabel 3 sebagai berikut.

Tabel 3.HasilPengujian*Command Execution*

Jenis	Hasil	Waktu Rata-Rata
Tanpa aplikasi pengaman web	Celah berhasil tereksekusi	0,0038
Menggunakan aplikasi pengaman web	inputan di tolak	0,0053

Dari pengujian pada tabel 3 didapatkan bahwa celah *Command Execution* dapat diblock oleh aplikasi pengaman web. Berikut contoh log yang di dapat dari pengujian ini:

"Command Injection |36.81.17.51 |2017-07-16 |16:44:1 4 |/Web-Firewall/ ?pattern=/etc/*&sort=name".

5. Pengujian Pengaman Celah Arbitrary File Upload
   Pengujian dilakukan pada fitur unggah pada web dummies. Dalam kasus ini extention yang di izinkan adalah jpg, jpeg, png dan gif. Adapun file yang akan di unggah memiliki extention php. Berikut hasil uji yang telah di lakukan dapat di lihat pada tabel 4 berikut ini.

Tabel 4. HasilPengujian Arbitrary File Upload

Jenis	**Hasil**	**Waktu Rata-Rata**
Tanpa aplikasi pengaman web	Celah berhasil tereksekusi	0,0043
Menggunakan aplikasi pengaman web	inputan di tolak	0,0054

Dari hasil diatas menujukan bahwa web tanpa aplikasi pengaman web dapat mengunggah berbagai file *executable* dan web dengan aplikasi pengaman web berhasil menolak unggahan yang berbahaya. Log yang di dapat ketika melakukan pengujian adalah

"Arbitrary File Upload|::1|2017-07-17|01:56:30 |/web-firewall/?p=fileu"

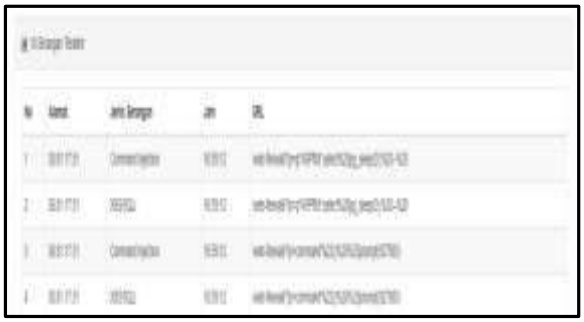
5.1 Laporan Penyerangan

Dari Log Serangan yang telah ada, aplikasi pengaman web akan membuat laporan lokasi IP penyerangang berbentuk map dan menampilkan 10 serangan, dapat di lihat pada gambar 3 berikut ini.

Gambar 3.Contoh Laporan Penyerangan

Pada gambar 3 terdapat 4 jenis laporan yaitu Serangan hari ini, IP terblokir hari ini Total IP terblokir dan total serangan.

Kemudian pada gambar 4 menunjukan 10 serangan terakhir sehingga admin dapat memantau serangan



Gambar 4. Contoh Tabel Laporan

Terakhir, terdapat juga laporan lokasi IP penyerang dengan google map, dapat di lihat pada gambar 5.

Gambar 5.Contoh Lokasi IP Penyerang

6. KESIMPULAN

Berdasarkan serangkaian proses mulai dari perancangan hingga implementasi pada aplikasi ini, didapatkan beberapa kesimpulan, antara lain :

1. Aplikasi pengaman web ini telah dapat digunakan untuk membantu mengamankan web dar iserangan-serangan berbahaya dengan cara filter untukcelahSQLidan XSS, pengenalan *string* untuk *Command Execution* dan *Arbitrary File Upload* serta pengenalan pola untuk *Brute Force*.
2. Dari hasil pengujian menunjukan aplikasi pengaman web ini dapat di andalkan untuk membantu mengamankan web dengan tingkat akurasi 100% dari celah yang telah ditentukan.

DAFTAR PUSTAKA

[1] Chandrika M (2013. Pencari celah keamanan pada aplikasi web, http://digilib.its.ac.id/ITS-paper-51021130002674/25617, diakses 10 Januari 2017

[2] Suluh Sri. (2009). Pembuatan framework audit situs web untuk auditor menggunakan model kuantitas perangkat lunak universitas tadulako.

[3] Wahyudi E F. (2013) . Studi kasus analisis keamanan sistem informasi perpustakaan. STMIK AKAKOM Yogyakarta

[4] Wahyuni S. (2010). Wirelees Aplication protocol Universitas Amikom Yogyakarta

[5] Gilmore , Jason. (2008). Secure PHP Programming. New York.

[6] Setia S Bella. (2012). Membangun aplikasi pembelajaran secure web programming berbasis owsp top 10. STMIK AKAKOM Yogyakarta

[7] Syaiful A.M (2014). encoding parameter untuk menangkal serangan, STMIK AKAKOM Yogyakarta